

# **User Guide for Q-par PO reflector analysis software (QPORAS)**

**Q-par/QPORAS/TR-PO1/0.2**



**Q-par Angus Ltd**

**IDEAS ENGINEERED**

**Cover + viii + 44 pages**

**May 2010**

Barons Cross Laboratories,  
Leominster, Herefordshire, HR6 8RS, UK.  
Tel: +44 (0) 1568 612138 Fax: +44 (0) 1568 616373  
Web: [www.q-par.com](http://www.q-par.com) E-mail: [sales@q-par.com](mailto:sales@q-par.com)



© Copyright 2010  
Q-par Angus Ltd., U.K.

Although the author has made every reasonable effort to ensure the accuracy of the contents of this document, neither the author nor Q-par Angus Ltd. will be held liable for damages resulting from the use or application of the material contained, to the extent permitted by international law.

This document may be distributed freely subject to the following conditions:

The document may not be changed or modified or the contents altered. It must remain complete without selective removal of material. It may not be sold or reproduced for commercial gain, without the written approval of an authorised representative of Q-par Angus Ltd. (UK).

Q-par Angus Ltd.  
Barons Cross Lodge, Leominster,  
Herefordshire, HR6 8RS, UK.

<b>Author</b>	Dr A. J. Mackay
<b>Date</b>	May 2010
<b>Issued by</b>	Q-par Angus Ltd. Barons Cross Laboratories Leominster Herefordshire HR6 HRS UK.

## Document changes record

Issue	Date	Change summary
Issue 0.1	September 2008	In progress
Issue 0.2	May 2010	Extended

## **Abstract**

This guide provides a technical overview and description of the use of the Q-par Physical Optics Reflector Analysis Software (QPORAS). The software uses a meshed representation of a general reflector under illumination by a plane wave or feed to provide gain and radiation patterns. It is based on the use of Physical Optics applied to a triangulated regular meshed reflector of arbitrary shape.

## List of contents

<b>Document changes record</b>	<b>iv</b>
<b>Abstract</b>	<b>v</b>
<b>List of contents</b>	<b>vi</b>
<b>List of figures</b>	<b>viii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background	1
1.2 Overview of the physical model	1
<b>2 The mathematical formulation</b>	<b>3</b>
2.1 The scattered far field	3
2.2 Scattered field directions and definitions	5
2.3 The Physical Optics integration	8
2.4 The incident fields	11
2.5 The scattered near field	13
2.6 Blockage effects	14
<b>3 Use of the software</b>	<b>15</b>
3.1 Introduction	15
3.2 The <i>CALCOPTS</i> key word	18
3.3 The <i>FILENAME</i> key word	19
3.4 The <i>PLOTFILE</i> key word	20
3.5 The <i>ANGLES</i> key word	21
3.6 The <i>ANGLECUT</i> key word	22
3.7 The <i>FREQS</i> key word	22
3.8 The <i>FEEDCEN</i> key word	23
3.9 The <i>FEEDROT</i> key word	23
3.10 The <i>PLANEWAVE</i> key word	24
3.11 The <i>DIPOLE</i> key word	25
3.12 The <i>RECTHORN</i> key word	25
3.13 The <i>GEOMFILE</i> key word	28
3.14 The <i>SURFACE</i> key word	30
3.15 The <i>BOUNDARY</i> key word	35
3.16 The <i>BLOCKAGE</i> key word	37

3.17	The <i>NEARFIELD</i> key word	38
3.18	The <i>GRASPOUT</i> key word	40
3.19	The <i>FILEREF</i> key word	40
3.20	The <i>FEEDFILE</i> key word	41
3.21	The <i>FARPOL</i> key word	43
<b>4</b>	<b>References</b>	<b>44</b>

## List of figures

2-1	Definition of surface normal	4
2-2	Arbitrary cuts on the surface of a sphere, specified by <i>ANGLECUT</i>	6
3-1	Main reflector global coordinate system	16
3-2	The feed coordinate system and Euler angles	16
3-3	Aperture excitation types for a rectangular horn	26
3-4	Boundary rectangle and its projection	36
3-5	Near field region, shadow masking and far field	38

# 1 Introduction

## 1.1 Background

Although there are many powerful and efficient software packages available for the accurate analysis of reflector antennas, these tend to be expensive commercial packages which are not readily affordable. For example a full version of GRASP [1], often cited as an industry standard, costs several tens of thousands of pounds. Unfortunately, free software is generally not available to analyse general shaped reflectors with standard feeds.

This report documents our own in-house software (which, in this report, we will call QPORAS) intended for this purpose. This is not intended to compete with GRASP or similar general purpose analysis software, but it is adaptable and can be added and modified as and when desired. It is written in FORTRAN for use on a Linux platform, but may be ported to other systems since no system-dependent calls are implemented.

## 1.2 Overview of the physical model

A general shaped curved reflecting surface is assumed topologically open with no part shadowing any other part under a point illuminator or incident plane wave. The surface can be described in several different ways either accepting a previously meshed structure or meshing the structure itself. The mesh is regular and composed of triangular facets.

Unlike moment method or finite element analysis methods there are no hard restrictions on how large each facet may be compared to a wavelength. The mesh size is primarily governed by geometrical requirements. For example, a flat rectangular plate reflector could be modelled by only two triangular facets with no loss of accuracy, independent of the wavelength. However, any representation of a curved surface by a faceted (polygonal) surface involves a difference in its electromagnetic properties so in such cases the maximum departure of a flat triangle from the curved surface should be small compared to a wavelength. It is still therefore a rule for a general curved surface that a triangle mesh cell should be small compared to a wavelength.

The main faceted reflector is assumed to be perfectly conducting (as of version 0.0.12) modelled using the version of physical optics that assumes that the induced electric currents at a point  $\underline{r}'$  on the surface,  $\mathcal{S}$ ,  $\underline{J}_e(\underline{r}')$  is given by the approximation,

$$\underline{J}_e(\underline{r}') = 2\hat{n}(\underline{r}') \times \underline{H}_{inc}(\underline{r}') \text{ for } \underline{r}' \in \mathcal{S} \quad (1-1)$$

where  $\hat{n}(\underline{r}')$  is an outward pointing surface normal and  $\underline{H}_{inc}(\underline{r}')$  is the incident magnetic field.

This physical optics (PO) assumption is an approximation which is generally considered accurate if, These requirements are not necessarily independent and may sometimes be relaxed under special conditions. Here they should be considered as rules of thumb. It is,

1. The radius of curvature of the surface is everywhere large compared to a wavelength except possibly over a region which is small compared to the rest of the structure.
2. Every point of the surface is under direct illumination of the source without self-shadowing of one part of the structure by another.
3. The incident field is locally planar.
4. The angle of incidence made between a locally planar wave and the surface normal should not be close to grazing incidence except possibly over a region which is small compared to the rest of the structure.
5. The surface should be much larger than a wavelength.

in fact, extremely difficult to obtain precise conditions that determine the accuracy of the physical optics method and under what conditions. To our knowledge, this remains an open question in mathematical physics.

In the discussion to date, *shadowing* refers to a purely geometrical effect. This is properly defined in the high frequency limit in terms of ray tracing. If a source is represented by a wavefront which is modelled by a collection of ray bundles, then if a given ray intersects the surface more than once the surface at the second intersection point is said to be in the shadow of the surface at the first intersection point. Shadowing, within physical optics, can only be modelled by making secondary approximations. Generally (though not always) this is accomplished by assuming an incident field that is defined by the value of the field within a ray bundle at the point of intersection with the surface. Thus, if a point of the surface is in shadow, the incident field is there assumed to be zero. In this approximation we assume sudden transitions of electric current; i.e. the electric current on the surface varies continuously until a region of shadow where the induced electric current drops to zero.

Although theoretically admissible, we always assume no self-shadowing of the main reflector. Shadowing is permissible only when considering blockage effects.

Various sources are permitted to illuminate the main reflector. These may be placed at any position and orientation. Currently (version 0.0.12) feed types include: a standard rectangular aperture horn, a dipole source, a plane wave illuminator.

The main purpose of the software is to compute radiation patterns in the far-field of the antenna. However there are applications where the near field is also required and, if there is blockage, a means to re-compute the far field from the (possibly blocked) near field. These are all options within the software.

## 2 The mathematical formulation

### 2.1 The scattered far field

The radiation pattern and gain are determined from the scattered far-field. We define the far-field electric field scattering coefficient by  $\underline{E}_s$ , where

$$\underline{E}_s = \lim_{r \rightarrow \infty} (k_0 r e^{jk_0 r} \underline{E}(r)) \quad (2-1)$$

where  $k_0$  is the free space wave number,  $r = |\underline{r}|$ ,  $\underline{r}$  is a point in space and  $\underline{E}(\underline{r})$  is the electric field at  $\underline{r}$ . The coordinate origin is assumed at the reflector and we may use a standard definition for  $\underline{E}_s$  (e.g. [1]) given by,

$$\begin{aligned} \underline{E}_s = & \frac{-jZ_0}{4\pi} \iint_{\mathcal{S}} (\underline{J}_e - [\underline{J}_e \cdot \hat{\underline{r}}] \hat{\underline{r}}) e^{jk_0 \underline{r}' \cdot \hat{\underline{r}}} k_0^2 ds' \\ & + \frac{j}{4\pi} \hat{\underline{r}} \times \iint_{\mathcal{S}} \underline{J}_m e^{jk_0 \underline{r}' \cdot \hat{\underline{r}}} k_0^2 ds' \end{aligned} \quad (2-2)$$

where  $Z_0$  is the free space impedance,  $\underline{r} = r\hat{\underline{r}}$  is the observation vector and  $\underline{r}'$  is a point in the surface  $\mathcal{S}$  over which the integration is performed.  $\underline{J}_e$  is the equivalent electric current and  $\underline{J}_m$  is the equivalent magnetic current on the surface defined by,

$$\underline{J}_e(\underline{r}') = \hat{\underline{n}} \times (\underline{H}_{inc} + \underline{H}_{ref} - \underline{H}_{tran}) \quad (2-3)$$

and

$$\underline{J}_m(\underline{r}') = -\hat{\underline{n}} \times (\underline{E}_{inc} + \underline{E}_{ref} - \underline{E}_{tran}) \quad (2-4)$$

where  $\hat{\underline{n}}$  is the outward pointing surface normal at the point  $\underline{r}'$  (see figure 2-1),  $\underline{H}_{inc}$  is the incident magnetic field in the absence of the scatterer,  $\underline{H}_{ref}$  is the scattered magnetic field just off the scatter in the direction of positive  $\hat{\underline{n}}$  and  $\underline{H}_{tran}$  is the transmitted magnetic field (non-zero only if the surface represents an interface between two regions of dielectric). Similar definitions apply to the electric field terms. Under the PO approximation for a perfect conductor,

$$\underline{J}_e = 2\hat{\underline{n}} \times \underline{H}_{inc} \quad (2-5)$$

$$\underline{J}_m = 0 \quad (2-6)$$

so far our purpose we may ignore the second integral term in (2-2)

It may be shown that the gain of the antenna, in the direction  $\hat{\underline{r}}$ , is given by

$$G = \frac{|\underline{E}_s|^2}{2k_0^2 Z_0} \cdot \frac{4\pi}{P_{inc}} \quad (2-7)$$

where  $P_{inc}$  is the incident power. For a feed, this may be defined as the power radiated by the feed. For an incident plane wave, this may be defined as the product of the projected area of the reflector in the direction of the incident plane wave and the intensity of the plane wave.

Since we assume the source field from the feed does not contribute to the scattered far field we will also assume that  $P_{inc}$  is the radiated power from the feed, radiated in all directions. This definition is especially significant for a dipole feed source.

We now assume that the curved surface can be approximately represented by a polygonal surface consisting of contiguous non-overlapping flat facets whose maximum deviation from the curved surface that they represent is small. Over the  $i_{th}$  such flat facet,  $\hat{n}(\underline{r}') = \hat{n}_i$  is a constant. In addition, if we further assume that each such facet is small enough to lie in the far field of the source then the direction of incidence does not vary over the facet and  $\underline{H}_{inc}(\underline{r}')$  may be approximated by,

$$\underline{H}_{inc}(\underline{r}') \approx \underline{H}_i e^{-j\hat{k}_i \cdot \underline{r}'} \text{ over the } i_{th} \text{ facet} \quad (2-8)$$

where  $\hat{k}_i = k_0 \hat{k}_i$ ,  $\hat{k}_i$  is the direction of the incident wave on the  $i_{th}$  facet and  $\underline{H}_i$  is the constant vector amplitude of the magnetic field over the  $i_{th}$  facet.

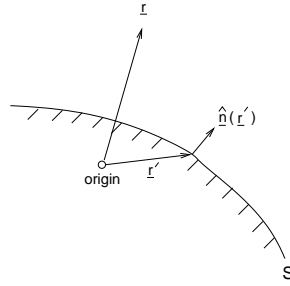


Figure 2-1: Definition of surface normal

It then follows that,

$$\underline{E}_s(\hat{k}_s) = \frac{-jZ_0 k_0^2}{4\pi} \sum_i \underline{J}_i^{(p)} \iint_{S_i} e^{jk_0 \underline{r}' \cdot (\hat{k}_s - \hat{k}_i)} dS' \quad (2-9)$$

where the projected current amplitude,  $\underline{J}_i^{(p)}$ , is constant over each facet;

$$\underline{J}_i^{(p)} = \underline{J}_i - [\underline{J}_i \cdot \hat{k}_s] \hat{k}_s \quad (2-10)$$

where

$$\underline{J}_i = 2\hat{n}_i \times \underline{H}_i \quad (2-11)$$

and  $\hat{k}_s = \hat{r}$  is the scatter direction. The sum over  $i$  is over all the flat facets,  $S_i$ , representing the surface.

## 2.2 Scattered field directions and definitions

The direction of the required scattered field is specified within the software using spherical polar coordinates,  $\theta$  and  $\phi$ . These angles are illustrated in figure 3-1. The field directions are specified with respect to the directions of increasing  $\theta$ , defined by the unit vector  $\hat{\theta}$  and the direction of increasing  $\phi$ , defined by the unit vector  $\hat{\phi}$ . Expressed in cartesian coordinates,

$$\begin{aligned}\hat{\theta} &= \cos \phi \cos \theta \hat{x} + \sin \phi \cos \theta \hat{y} - \sin \theta \hat{z} \\ \hat{\phi} &= -\sin \phi \hat{x} + \cos \phi \hat{y}\end{aligned}\quad (2-12)$$

We may write the components of the scattered electric field vector,

$$\underline{E}_s(\hat{k}_s) = E_1 \hat{\theta} + E_2 \hat{\phi} \quad (2-13)$$

The coefficients  $E_1$  and  $E_2$  are the (complex) scattering amplitudes in these directions. The program permits a rotation of this polarisation base by a constant angle  $\zeta_s$  so that final outputs may be expressed at an angle  $\zeta_s$  to the spherical angle directions, i.e. in the directions,

$$\begin{aligned}\hat{\theta}' &= \cos \zeta_s \hat{\theta} - \sin \zeta_s \hat{\phi} \\ \hat{\phi}' &= \sin \zeta_s \hat{\theta} + \cos \zeta_s \hat{\phi}\end{aligned}\quad (2-14)$$

Writing,

$$\underline{E}_s(\hat{k}_s) = E_1' \hat{\theta}' + E_2' \hat{\phi}' \quad (2-15)$$

we therefore have,

$$\begin{aligned}E_1' &= \cos \zeta_s E_1 - \sin \zeta_s E_2 \\ E_2' &= \sin \zeta_s E_1 + \cos \zeta_s E_2\end{aligned}\quad (2-16)$$

The default angle for  $\zeta_s = 0$ , but may be changed using the input tag command *FARPOL*, as given later. Note that the sense of the angle for  $\zeta_s$  is chosen so that a positive angle  $\zeta_s$  is equivalent to a positive rotation of the feed by an angle  $\psi_f$  in an axisymmetric system. Thus in a system where the reflector is axisymmetric and the feed lies on this axis, choosing  $\psi_f = \zeta_s$  implies that  $E_1'$  and  $E_2'$  remain invariant of  $\zeta_s$  when  $\theta = \phi = 90^\circ$ .

The key word *ANGLES* defines a block of required  $\theta$  and  $\phi$  angles directly. However, it is also required to specify an arbitrary angular cut which cannot be achieved for constant  $\theta$  or constant  $\phi$ . This is accomplished using the *ANGLECUT* key word. In this case, we define a particular direction specified by  $\theta = \theta_0$  and  $\phi = \phi_0$ . We then specify a scan direction angle,  $\eta_0$ , made with respect to the direction  $\hat{\theta}_0$  and vary the angle  $\nu$  along this cut. This is illustrated in figure 2-2 below.

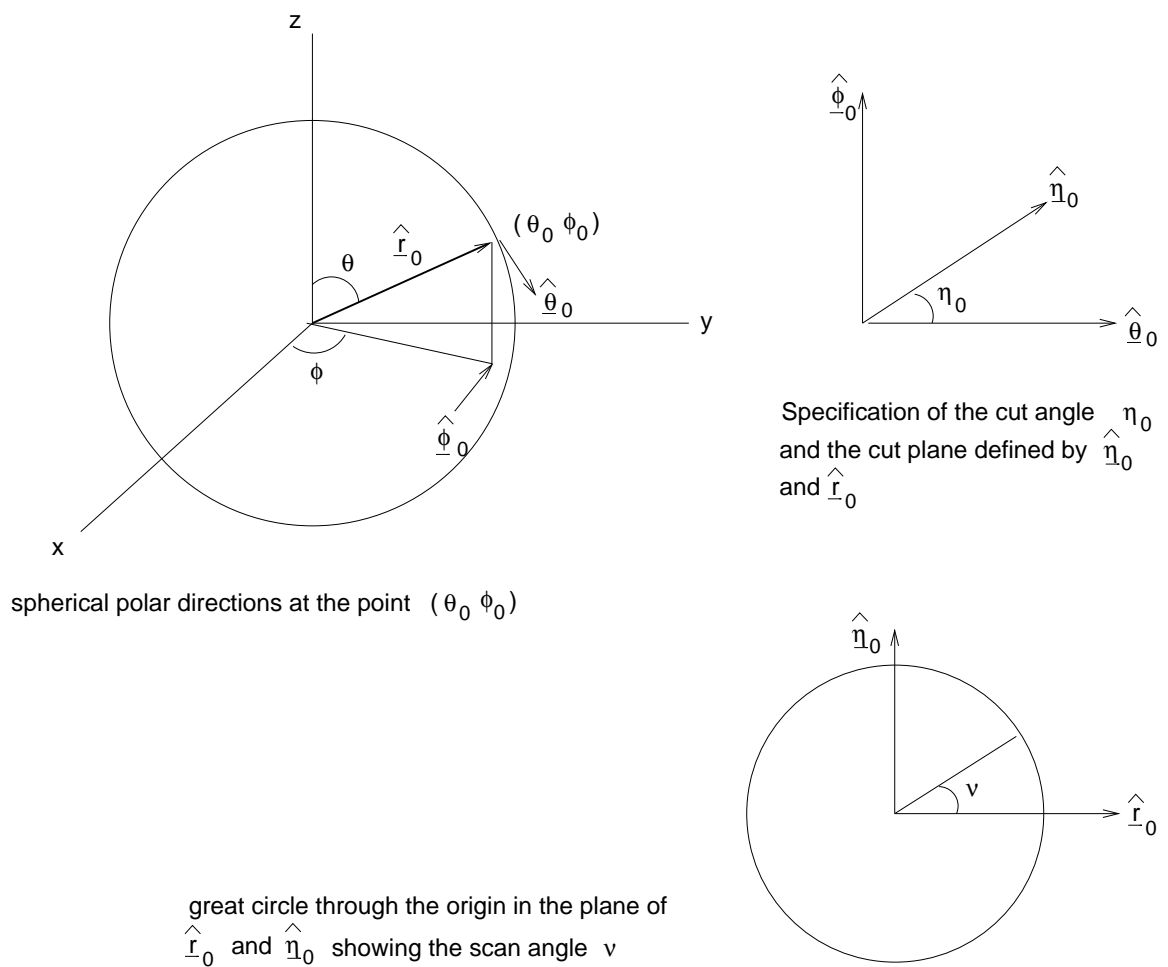


Figure 2-2: Arbitrary cuts on the surface of a sphere, specified by ANGLECUT

The angle  $\eta_0$  at the point  $(\theta_0, \phi_0)$  defines the direction,

$$\underline{\hat{\eta}}_0 = \cos \eta_0 \underline{\hat{\theta}}_0 + \sin \eta_0 \underline{\hat{\phi}}_0 \quad (2-17)$$

where

$$\begin{aligned} \underline{\hat{\theta}}_0 &= \cos \phi_0 \cos \theta_0 \underline{\hat{x}} + \sin \phi_0 \cos \theta_0 \underline{\hat{y}} - \sin \theta_0 \underline{\hat{z}} \\ \underline{\hat{\phi}}_0 &= -\sin \phi_0 \underline{\hat{x}} + \cos \phi_0 \underline{\hat{y}} \end{aligned} \quad (2-18)$$

We also require the direction vector at  $(\theta_0, \phi_0)$ , defined by,

$$\underline{\hat{r}}_0 = \cos \phi_0 \sin \theta_0 \underline{\hat{x}} + \sin \phi_0 \sin \theta_0 \underline{\hat{y}} + \cos \theta_0 \underline{\hat{z}} \quad (2-19)$$

The direction vector under the scan is now given by,

$$\underline{\hat{r}} = \cos \nu \underline{\hat{r}}_0 + \sin \nu \underline{\hat{\eta}}_0 \quad (2-20)$$

but we also have,

$$\underline{\hat{r}} = \cos \phi \sin \theta \underline{\hat{x}} + \sin \phi \sin \theta \underline{\hat{y}} + \cos \theta \underline{\hat{z}} \quad (2-21)$$

so equating terms, we obtain expressions for  $\theta$  and  $\phi$  in terms of  $\theta_0$ ,  $\phi_0$ ,  $\eta_0$  and  $\nu$ . The cartesian components of  $\underline{\hat{r}}$  are given by,

$$\begin{aligned} r_x(\nu) &= \cos \nu \cos \phi_0 \sin \theta_0 + \sin \nu \cos \eta_0 \cos \phi_0 \cos \theta_0 - \sin \nu \sin \eta_0 \sin \phi_0 \\ r_y(\nu) &= \cos \nu \sin \phi_0 \sin \theta_0 + \sin \nu \cos \eta_0 \sin \phi_0 \cos \theta_0 + \sin \nu \sin \eta_0 \cos \phi_0 \\ r_z(\nu) &= \cos \nu \cos \theta_0 - \sin \nu \cos \eta_0 \sin \theta_0 \end{aligned} \quad (2-22)$$

so,

$$\cos \theta = r_z \quad (2-23)$$

from which we may obtain  $\theta$ , under the assumption  $0 \leq \theta \leq \pi$ , and

$$\phi = \arctan_2(r_y, r_x) \quad (2-24)$$

where the  $\arctan_2$  function is the angle-unambiguous arc-tangent of the ratio  $r_y/r_x$ .

### 2.3 The Physical Optics integration

In evaluating the PO integrals, we assume the facets are triangular in which case the integrals are of Gorden's generic form,

$$I_G(\underline{w})_i = \iint_{\mathcal{A}_i} e^{jk_0(\underline{w} \cdot \underline{x})} d\underline{x} \quad (2-25)$$

over a general triangle  $\mathcal{A}_i$  with area  $A_i$ . If the triangle is defined by the coordinates  $\underline{x}_1$ ,  $\underline{x}_2$  and  $\underline{x}_3$  of the vertices, then it can be shown after some algebra that,

$$I_G = \frac{2Ae^{jk_0\underline{x}_1 \cdot \underline{w}}}{js_{31}} \left( \frac{e^{js_{31}}(e^{js_{23}} - 1)}{js_{23}} - \frac{(e^{js_{21}} - 1)}{js_{21}} \right) \quad (2-26)$$

where the  $i$  subscripts are no longer shown explicitly in the notation and

$$\begin{aligned} s_{21} &= k_0(\underline{x}_2 - \underline{x}_1) \cdot \underline{w} \\ s_{31} &= k_0(\underline{x}_3 - \underline{x}_1) \cdot \underline{w} \\ s_{23} &= k_0(\underline{x}_2 - \underline{x}_3) \cdot \underline{w} \end{aligned} \quad (2-27)$$

In the limit that one or more of the  $s$  terms tends to zero, the expression (2-26) must be modified to permit computation. To do this, define

$$P_0(\delta) = \int_0^1 e^{j\eta\delta} d\eta \quad (2-28)$$

$$P_1(\delta) = \int_0^1 \eta e^{j\eta\delta} d\eta \quad (2-29)$$

$$P_2(\delta) = \int_0^1 \eta^2 e^{j\eta\delta} d\eta \quad (2-30)$$

which may be evaluated by,

$$P_0(\delta) = \begin{cases} \frac{1}{j\delta}(e^{j\delta} - 1) & \text{for } \delta \neq 0 \\ (1 - \delta^2/6) + \frac{j\delta}{2}(1 - \delta^2/12) & \text{for } \delta \rightarrow 0 \end{cases} \quad (2-31)$$

$$P_1(\delta) = \begin{cases} \frac{e^{j\delta}}{j\delta} - P_0(\delta)/(j\delta) & \text{for } \delta \neq 0 \\ (1/2 - \delta^2/8) + j\delta/3 & \text{for } \delta \rightarrow 0 \end{cases} \quad (2-32)$$

$$P_2(\delta) = \begin{cases} \frac{e^{j\delta}}{j\delta} - 2P_1(\delta)/(j\delta) & \text{for } \delta \neq 0 \\ 1/3 + j\delta/4 & \text{for } \delta \rightarrow 0 \end{cases} \quad (2-33)$$

Now, for  $s_{31} \rightarrow 0$ ,

$$I_G = 2Ae^{jk_0\underline{x}_1 \cdot \underline{w}}[(1 - js_{31}/2)P_0(s_{21}) + (-1 + js_{31})P_1(s_{21}) - j(s_{31}/2)P_2(s_{21})] \quad (2-34)$$

For  $s_{31} \neq 0$  we must consider the cases  $s_{23} \rightarrow 0$  and/or  $s_{21} \rightarrow 0$  in which case,

$$I_G = \frac{2Ae^{jk_0\underline{x}_1 \cdot \underline{w}}}{js_{31}}[e^{js_{31}}P_0(s_{23}) - P_0(s_{21})] \quad (2-35)$$

The area of each triangle may be conveniently evaluated using Heron's formula,

$$A = \sqrt{s(s-a)(s-b)(s-c)} \quad (2-36)$$

where

$$s = \frac{a+b+c}{2} \text{ and } a = |\underline{x}_3 - \underline{x}_2|, b = |\underline{x}_3 - \underline{x}_1|, c = |\underline{x}_2 - \underline{x}_1| \quad (2-37)$$

If a surface is represented by triangles which are in turn defined by nodes representing the triangle vertices, there is a numerical dichotomy; the fields and directions of waves are naturally defined at the nodes, but they are required on the triangle surfaces where the surface normals are defined. However, since it is assumed that these quantities are slowly varying, using (2-8) we may take,

$$\underline{H}_i = \frac{1}{3}(e^{j\mathbf{k}_i \cdot \underline{x}_{i1}} \underline{H}_{inc}(\underline{x}_{i1}) + e^{j\mathbf{k}_i \cdot \underline{x}_{i2}} \underline{H}_{inc}(\underline{x}_{i2}) + e^{j\mathbf{k}_i \cdot \underline{x}_{i3}} \underline{H}_{inc}(\underline{x}_{i3})) \quad (2-38)$$

where  $\underline{x}_{ij}$  are the vertices  $\underline{x}_j$  of the  $i_{th}$  triangle. The incident wave directions  $\mathbf{k}_i$  must also be approximated. We permit two ways of doing this within the software, though in practise there is very little difference between them if our assumptions on slow variations are correct.

The first method assumes that the phase centre of the source is specified, and that the field radiated by the source may be considered to radiate from this phase centre at all points over the reflector surface. In this case, if the phase centre is specified at the coordinates  $\underline{r}_0$ , then the vector from the phase centre to the triangle centroid is given by,

$$\underline{r}_i^{(c)} = \frac{1}{3}(\underline{x}_{i1} + \underline{x}_{i2} + \underline{x}_{i3}) - \underline{r}_0 \quad (2-39)$$

hence

$$\underline{k}_i = k_0 \underline{r}_i^{(c)} / |\underline{r}_i^{(c)}| \quad (2-40)$$

The second method makes no assumption on the position of the phase centre, but assumes the wave direction is that of the Poynting vector. This basically assumes that the field is locally planar with the E and H-fields orthogonal and in-phase. The direction of the wave at a general point in space may be defined by,

$$\hat{\underline{k}}^{(p)}(\underline{x}) = \frac{\Re(\underline{E}(\underline{x}) \times \underline{H}^*(\underline{x}))}{|\Re(\underline{E}(\underline{x}) \times \underline{H}^*(\underline{x}))|} \quad (2-41)$$

which is required when  $\underline{E}(\underline{x}) = \underline{E}_{inc}(\underline{x})$  and  $\underline{H}(\underline{x}) = \underline{H}_{inc}(\underline{x})$ . Evaluation of the incident fields at the nodal coordinates depends on the method employed (see below).

Again, since the fields are only available at the nodes, we employ an average defined by,

$$\underline{k}_i = \frac{k_0(\hat{\underline{k}}^{(p)}(\underline{x}_{i1}) + \hat{\underline{k}}^{(p)}(\underline{x}_{i2}) + \hat{\underline{k}}^{(p)}(\underline{x}_{i3}))}{|\hat{\underline{k}}^{(p)}(\underline{x}_{i1}) + \hat{\underline{k}}^{(p)}(\underline{x}_{i2}) + \hat{\underline{k}}^{(p)}(\underline{x}_{i3})|} \quad (2-42)$$

Finally, the surface normals  $\hat{n}_i$  must be computed. There is a possible sign ambiguity that must be resolved since we require the surface normals always to be pointed in the “outward” direction, remembering that the surface interface represents the surface of a solid region of conductor. We define this such that  $\hat{n}_i \cdot \underline{k}_i < 0$  for all points on the scattering surface. Note that transition regions, where there may be a required change in sign of  $\hat{n}_i \cdot \underline{k}_i$ , are ruled out by our assumption that no self-shadowing of the reflector is permitted.

We therefore define,

$$\underline{\hat{n}}'_i = \frac{(\underline{x}_{i2} - \underline{x}_{i1}) \times (\underline{x}_{i3} - \underline{x}_{i2})}{|(\underline{x}_{i2} - \underline{x}_{i1}) \times (\underline{x}_{i3} - \underline{x}_{i2})|} \quad (2-43)$$

and

$$\underline{\hat{n}}_i = \begin{cases} \underline{\hat{n}}'_i & \text{if } \underline{\hat{n}}'_i \cdot \underline{r}_i^{(c)} < 0 \\ -\underline{\hat{n}}'_i & \text{if } \underline{\hat{n}}'_i \cdot \underline{r}_i^{(c)} > 0 \end{cases} \quad (2-44)$$

## 2.4 The incident fields

In the previous section it was assumed that the incident fields,  $\underline{E}_{inc}(\underline{x})$  and  $\underline{H}_{inc}(\underline{x})$  are available at the coordinates specifying the triangular nodes. The software has various feed options which calculate or estimate these fields in different manners. An electric dipole may be specified using the *DIPOLE* key word, or a plane wave using the *PLANEWAVE* key word which provide an analytic determination of these fields. More generally, however, no analytic formulation is available.

A rectangular horn or a user specified aperture distribution may be provided, too. The latter allows either the electric field or both the electric and the magnetic field to be given on a plane in space. Once these aperture fields are specified the far-fields  $\underline{E}_{inc}(\underline{x})$  and  $\underline{H}_{inc}(\underline{x})$  may be determined by integration.

The basic formulas for the E and H radiated fields, evaluated at a point  $\underline{R}_p$ , in terms of the aperture fields  $\underline{E}_r$  and  $\underline{H}_r$  are given by Silver (section 5-11, of [5]),

$$\underline{E}(\underline{R}_p) = \frac{1}{4\pi j\omega\epsilon_0} \int_A (k_0^2(\underline{n} \times \underline{H}_r)\Psi + (\underline{n} \times \underline{H}_r) \cdot \nabla(\nabla\Psi) + j\omega\epsilon_0(\underline{n} \times \underline{E}_r) \times \nabla\Psi) ds \quad (2-45)$$

and

$$\underline{H}(\underline{R}_p) = \frac{-1}{4\pi j\omega\mu_0} \int_A (k_0^2(\underline{n} \times \underline{E}_r)\Psi + (\underline{n} \times \underline{E}_r) \cdot \nabla(\nabla\Psi) - j\omega\mu_0(\underline{n} \times \underline{H}_r) \times \nabla\Psi) ds \quad (2-46)$$

where the surface integral is over the aperture plane,  $A$ , and  $\underline{E}_r(\underline{x}_s)$  and  $\underline{H}_r(\underline{x}_s)$  are evaluated on this plane and  $\underline{n}(\underline{x}_s)$  is the outward pointing surface normal in the direction of the radiating field, taken as a constant unit vector for this application.  $\epsilon_0$  and  $\mu_0$  are the permittivity and permeability of free space.  $\Psi$  is defined by,

$$\Psi(\underline{R}_p, \underline{x}_s) = \frac{e^{-jk_0r}}{r} \quad (2-47)$$

where

$$r^2 = |\underline{R}_p - \underline{x}_s|^2 \quad (2-48)$$

Notice that only the transverse fields with components in the plane normal to  $\underline{n}$  are required in the formulation. Without loss of generality, assume  $\underline{n} = \hat{\underline{z}}$  where  $\hat{\underline{z}}$  is the unit vector normal to the plane surface and we have,

$$\underline{n} \times \underline{E}_r = \hat{\underline{z}} \times \underline{E}_t \quad (2-49)$$

and

$$\underline{n} \times \underline{H}_r = \hat{\underline{z}} \times \underline{H}_t \quad (2-50)$$

where  $\underline{E}_t$  and  $\underline{H}_t$  are the tangential field components of the electric and magnetic fields. Specification of  $\underline{E}_t$  and  $\underline{H}_t$  depends on the feed method, as outlined shortly.

In (2-45) and (2-46) the “ $\nabla$ ” gradient terms are interpreted as acting on  $r$ , i.e.

$$\nabla \Psi = \nabla (e^{-jk_0 r}/r) = \left( jk_0 + \frac{1}{r} \right) \frac{e^{-jk_0 r}}{r} \hat{r} \quad (2-51)$$

and

$$(\underline{J} \cdot \nabla) \nabla \Psi = \left( -k_0^2 (\underline{J} \cdot \hat{r}) \hat{r} + \frac{3}{r} (jk_0 + 1/r) (\underline{J} \cdot \hat{r}) \hat{r} - \frac{\underline{J}}{r} (jk_0 + 1/r) \right) \frac{e^{-jk_0 r}}{r} \quad (2-52)$$

where

$$\hat{r} = \frac{\underline{R}_p - \underline{x}_s}{|\underline{R}_p - \underline{x}_s|} \quad (2-53)$$

For a rectangular feed, we assume a feed which flares from waveguide with dimensions  $d_x$  by  $d_y$  to an aperture of dimensions  $D_x$  by  $D_y$  over a length  $H$ , where  $D_x > d_x$  and  $D_y > d_y$ . The flare introduces a phase distortion. We assume operation in either the TE01 or TE10 mode in which case the aperture field takes the approximate form,

$$\underline{E}_t = E_a \hat{y} \cos \left( \frac{\pi x}{D_x} \right) \exp \left( -j(\alpha(2x/D_x)^2 + \beta(2y/D_y)^2) \right) \quad (2-54)$$

or the alternative mode with interchange of  $x$  and  $y$  coordinates. The values for  $\alpha$  and  $\beta$  may be approximated by,

$$\begin{aligned} \alpha &= k_0 (\sqrt{r_x^2 + (D_x/2)^2} - r_x) \\ \beta &= k_0 (\sqrt{r_y^2 + (D_y/2)^2} - r_y) \end{aligned} \quad (2-55)$$

where

$$r_x \approx r_y \approx H \quad (2-56)$$

The coefficient  $E_a$  is a normalisation constant which is adjusted so that the feed always radiates exactly 1 watt. We assume that the admittance at the aperture is approximately that of free space, in which case it can be shown that,

$$E_a = \frac{2}{\sqrt{D_x D_y}} \left( \frac{\epsilon_0}{\mu_0} \right)^{-1/4} \quad (2-57)$$

The magnetic field and the electric field are related in a waveguide by the modal admittance. At the aperture of a flared horn this relation is only approximate. We have attempted to use numerical differentiation on the assumed electric field to generate a more consistent approximation but generally this seems to be less reliable. Our most robust approximation is simply to assume that at the aperture,

$$\underline{H}_t = \hat{z} \times \underline{E}_t / Z_0 \quad (2-58)$$

using the free space admittance. More accurate field distributions require the export of aperture field distribution files, e.g. from CST [4] using the *FEEDFILE* option.

When importing aperture field distributions we have two options; either we specify both the electric and magnetic fields in which case the approximation (2-58) is not made, or we specify the E-field and use (2-58) to estimate the H-field on the aperture.

Finally, we note that in the formulation (2-45) is not required to generate the electric field scattered by the reflector in (2-9), only (2-46). However we calculate both since the overhead in doing so is small and this allows us to determine Poynting vectors and local directions of incidence  $\underline{k}_i$  (the wave is assumed locally planar over a facet of the reflector) when the phase centre of the source field is not defined.

## 2.5 The scattered near field

We assume that there are no dielectrics present between the perfectly conducting reflector and a required set of points where the scattered near field is required. Let  $\underline{r}$  be an observation point where the field is required and let  $\mathcal{S}$  be the scatterer (reflector). The scattered electric field  $\underline{E}(\underline{r})$  is given (see, e.g. [1]) by,

$$\begin{aligned} \underline{E}(\underline{r}) = \frac{Z_0 k_0^2}{4\pi} \iint_{\mathcal{S}} \left[ \underline{J}_e(\underline{r}') \left( -\frac{j}{k_0 R} - \frac{1}{k_0^2 R^2} + \frac{j}{k_0^3 R^3} \right) \right. \\ \left. + \hat{R}(\underline{J}_e \cdot \hat{R}) \left( \frac{j}{k_0 R} + \frac{3}{k_0^2 R^2} - \frac{3j}{k_0^3 R^3} \right) \right] e^{-jk_0 R} d\mathcal{S}' \end{aligned} \quad (2-59)$$

where,  $R = |\underline{r} - \underline{r}'|$  and  $\hat{R} = (\underline{r} - \underline{r}')/R$ .

Similarly, the magnetic field is given by,

$$\underline{H}(\underline{r}) = \frac{k_0^2}{4\pi} \iint_{\mathcal{S}} \underline{J}_e(\underline{r}') \times \hat{R} \frac{1 + jk_0 R}{k_0^2 R^2} e^{-jk_0 R} d\mathcal{S}' \quad (2-60)$$

Again, under the PO approximation, we assume

$$\underline{J}_e = 2\hat{n}(\underline{r}') \times \underline{H}_{inc}(\underline{r}') \text{ for } \underline{r}' \in \mathcal{S} \quad (2-61)$$

Using the same faceted representation of the surface as above, we assume that each facet is small enough to be (individually) in the far-field of a fictitious elemental dipole source at  $\underline{r}$ . We may then safely assume that  $R$  varies very little over any given facet and all terms other than the phase variation involving  $R$  may be taken outside the integral. It may then be shown that,

$$\underline{E}(\underline{r}) = \frac{Z_0 k_0^2}{4\pi} \sum_i I_G(\underline{w})_i \underline{Q}_i e^{-jk_0(R_i^{(c)} + \hat{R}_i^{(c)} \cdot \underline{x}_i^{(c)})} \quad (2-62)$$

where

$$\underline{w} = \hat{R}_i^{(c)} - \underline{k}_i \quad (2-63)$$

$$\hat{R}_i^{(c)} = \underline{R}_i^{(c)} / R_i^{(c)} \quad (2-64)$$

$$R_i^{(c)} = |\underline{R}_i^{(c)}| \quad (2-65)$$

$$\underline{R}_i^{(c)} = \underline{r} - \underline{r}_i^{(c)} \quad (2-66)$$

where  $I_G$ ,  $\hat{k}_i = \underline{k}_i/k_0$  and  $\underline{r}_i^{(c)}$  are defined in the previous section. The vector coefficient  $\underline{Q}_i$  is given by,

$$\underline{Q}_i = \underline{J}_i \left( -\frac{j}{k_0 R_i^{(c)}} - \frac{1}{k_0^2 (R_i^{(c)})^2} + \frac{j}{k_0^3 (R_i^{(c)})^3} \right) + \hat{\underline{R}}_i^{(c)} (\underline{J}_i \cdot \hat{\underline{R}}_i^{(c)}) \left( \frac{j}{k_0 R_i^{(c)}} + \frac{3}{k_0^2 (R_i^{(c)})^2} - \frac{3j}{k_0^3 (R_i^{(c)})^3} \right) \quad (2-67)$$

Similarly, it may be shown that

$$\underline{H}(\underline{r}) = \frac{k_0^2}{4\pi} \sum_i I_G(\underline{w})_i \underline{P}_i e^{-jk_0(R_i^{(c)} + \hat{\underline{R}}_i^{(c)} \cdot \underline{r}_i^{(c)})} \quad (2-68)$$

where the vector coefficient,

$$\underline{P}_i = \left( \frac{1 + jk_0 R_i^{(c)}}{k_0^2 (R_i^{(c)})^2} \right) \underline{J}_i \times \hat{\underline{R}}_i^{(c)} \quad (2-69)$$

## 2.6 Blockage effects

There are a large number of logically and computationally different methods for approximating the effect of blockages under high frequency and/or physical optics approximations. In our implementation we employ a combination of ray-tracing and PO on the main reflector, avoiding the need to recompute far fields at other than the main reflector. In outline, we assume a general blocking surface either specified by another triangulated surface represented as a user specified “points and joins” file or that the feed aperture is blocking the reflected wave. This latter option is relevant only to certain feed types (i.e. it is not relevant to an infinitesimally small dipole or incident plane wave).

The primary method follows:

This is one of the simplest and fastest algorithms to establish the effects of blockage. It is probably also one of the least accurate and must be treated with care until we are able to establish the level of accuracy that may be expected. A second method is also implemented, which requires the computation of the near field. This is described in more detail in the user section on the *NEARFIELD* command.

1. Determine the incident wave direction  $\hat{k}_i$  at the centroid of each facet.
2. Determine the reflected wave direction  $\hat{k}_r$  at the centroid of each facet.
3. Trace a ray from the centroid of each facet with wave direction  $\hat{k}_r$ .
4. Determine whether any such reflected ray intersects any part of the blocking surface.
5. If such a ray does intersect, mark the facet on the reflector, from which the ray originates, as “shadowed”. It is assumed that the energy associated with this ray is totally absorbed, i.e. that the blocking structure is “black”.
6. Assume any “shadowed” facet does not contribute to the far field at any scatter angle or at any frequency. Exclude “shadowed” facets from the PO sum over facets.

### 3 Use of the software

#### 3.1 Introduction

The software is driven from an input script file using a command line console. The name of the executable is arbitrary, so let us assume here it is called **testpo**. The command would thus read:

**testpo <filename>**

where **<filename>** is the name of the script file. The program searches for recognised commands designated by a key word followed by a list of parameters. Unrecognised words are ignored, but it is good practise to comment out any unused commands with a symbol such as ‘%’. Generally (and as of version 0.0.12) of the software, any commands may be entered in any order.

The commands specify the calculation options, required frequencies and scan angles, geometry designations, coordinate rotations, etc. There are two coordinate systems used within the software. The first is the “global coordinate system” in which the main reflector is defined and to which the scan angles are referenced. Currently (as of version 0.0.12) the commands that define standard reflector shapes and their boundaries assume a reflector that is pointing forwards in the direction of the y-axis. This is illustrated in figure 3-1, together with the definitions of elevation scan angle  $\theta$  and azimuth angle  $\phi$ . There are commands, however, to read in surfaces defined by an arbitrary list of coordinates.

There also exists a separate coordinate system which defines how the feed is orientated. This is not relevant to an incident plane wave, which is specified in the global coordinate system.

The feed coordinate system is specified by the coordinates  $x'$ ,  $y'$  and  $z'$  such that the feed aperture is normal to the local  $z'$  axis. The nominal phase centre of the feed is assumed to lie on the origin of the primed coordinate system. The orientation of the feed coordinate system with respect to the global coordinate system is controlled by user-specified Euler angles,  $\theta_f$ ,  $\phi_f$  and  $\psi_f$ . This is illustrated in figure 3-2. The illustration of the Euler angles

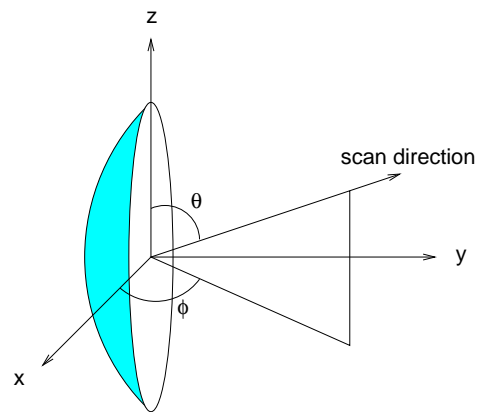


Figure 3-1: Main reflector global coordinate system

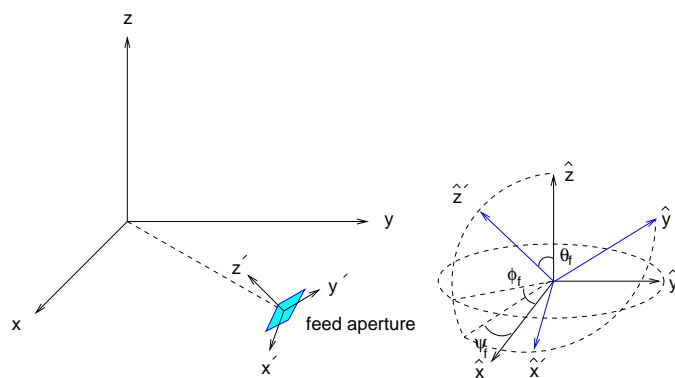


Figure 3-2: The feed coordinate system and Euler angles

is a little difficult; we define them as in [2]. If  $\underline{\hat{x}}$ ,  $\underline{\hat{y}}$  and  $\underline{\hat{z}}$  are unit vectors in direction of increasing  $x$ ,  $y$  and  $z$  and similarly,  $\underline{\hat{x}'}$ ,  $\underline{\hat{y}'}$  and  $\underline{\hat{z}'}$  are unit vectors in direction of increasing  $x'$ ,  $y'$  and  $z'$ , we have

$$\begin{aligned}
\underline{\hat{x}'} &= (-\sin \phi_f \sin \psi_f + \cos \theta_f \cos \phi_f \cos \psi_f) \underline{\hat{x}} \\
&\quad + (\cos \phi_f \sin \psi_f + \cos \theta_f \sin \phi_f \cos \psi_f) \underline{\hat{y}} \\
&\quad - \sin \theta_f \cos \psi_f \underline{\hat{z}} \\
\underline{\hat{y}'} &= (-\sin \phi_f \cos \psi_f - \cos \theta_f \cos \phi_f \sin \psi_f) \underline{\hat{x}} \\
&\quad + (\cos \phi_f \cos \psi_f - \cos \theta_f \sin \phi_f \sin \psi_f) \underline{\hat{y}} \\
&\quad + \sin \theta_f \sin \psi_f \underline{\hat{z}} \\
\underline{\hat{z}'} &= \sin \theta_f \cos \phi_f \underline{\hat{x}} + \sin \theta_f \sin \phi_f \underline{\hat{y}} + \cos \theta_f \underline{\hat{z}}
\end{aligned} \tag{3-1}$$

If a point in the global coordinate system is specified by  $\underline{r} = (x, y, z)$  and in the feed coordinate system by  $\underline{r}' = (x', y', z')$  then the transformation between the two is given by,

$$\begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix} = \underline{\underline{R}} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \tag{3-2}$$

where  $\underline{r}_0 = (x_0, y_0, z_0)$  is the position of the phase centre of the feed and the Euler rotation matrix  $\underline{\underline{R}}$  has coefficients with rows given by the row terms in (3-1).

A typical script file might take the form;

```

CALCOPTS 1
ANGLES 90.0, 0.0, 1, -0.0, 0.25, 361
FREQS 77000.0 0.0 1
FILENAME test_out junk_out
FEEDCEN 0.0 0.15 0.0
%%FEEDROT 0.000 0.000 0.0
%%RECTHORN 0.012 0.012 0.0300 0.0300 Y 0.10
PLANEWAVE 90.0 -45.0 0.0 0.0
PLOTFILE junk.ps 90.00 45.0 180.0 0.4 0.2 0.0 T F F F
SURFACE SPLINE1D spline1.dat
BOUNDARY ellipse 0.106, 0.075, 0.0 0.00 00.0 0.001
GEOMFILE qqj_pj.dat RW

```

where we have assumed illumination by a plane wave. The instructions to define a rectangular feed horn and its Euler rotation angles have been commented out. A definition of all the instructions now follows.

### 3.2 The *CALCOPTS* key word

#### Description

Defines one of several types of calculation options in the method of evaluation of the incident fields at the reflector. Currently (version 0.0.12) there are just two options. Option 1 uses both the incident electric and magnetic fields at each point on the reflector to determine the apparent direction of incidence using the Poynting vector method as described earlier. Option 2 uses only the incident electric fields together with a knowledge of the phase centre (or in the case of a plane wave, the defined direction of incidence) to determine the direction of incidence. The list below summarises its use.

Number of parameters	1
Key word requirement	Optional. May occur anywhere in a composite. If not present, Option 1 is assumed.
Number of occurrences	At most one.
File location	Anywhere within input file.
Parameter #1	Integer. Its value is 1 for option 1 and 2 for option 2.

### 3.3 The *FILENAME* key word

#### Description

Defines the name of the main output files. Currently (version 0.0.12) this takes two filenames, though only the first is used. (The second is reserved for further software enhancements). The first filename is the name of the output file containing the predicted far-field gains.

The list below summarises its use.

Number of parameters	2
Key word requirement	Mandatory.
Number of occurrences	One.
File location	Anywhere within input file.
Parameter #1	Character string containing no spaces or separators. Name of output gain file.
Parameter #2	Character string containing no spaces or separators. Currently unused.

### 3.4 The *PLOTFILE* key word

#### Description

Defines the name of optional output POSTSCRIPT files showing the reflector and feed. Intended for diagnostic purposes. There may be any number of these. The parameters control the Euler view angles and magnification as well as some logical flags to switch on and off various features.

The list below summarises its use.

Number of parameters	11
Key word requirement	Optional.
Number of occurrences	Any number.
File location	Anywhere within input file.
Parameter #1	Character string containing no spaces or separators. Name of output POSTSCRIPT file.
Parameter #2	Real number, $\theta_v$ . The $\theta$ – view rotation angle in global coordinate system, specified in degrees.
Parameter #3	Real number, $\phi_v$ . The $\phi$ – view rotation angle in global coordinate system, specified in degrees.
Parameter #4	Real number, $\psi_v$ . The $\psi$ – view rotation angle in global coordinate system, specified in degrees.
Parameter #5	Real number, magnification (scale) factor.
Parameter #6	Real number, x-coordinate display offset in meters.
Parameter #7	Real number, y-coordinate display offset in meters.
Parameter #8	Logical ‘T’ or ‘F’. If ‘T’ displays reflector.
Parameter #9	Logical ‘T’ or ‘F’. If ‘T’ displays shadow structure (when there is a blockage).
Parameter #10	Logical ‘T’ or ‘F’. If ‘T’ displays incident wave directions on each facet of reflector.
Parameter #11	Logical ‘T’ or ‘F’. If ‘T’ displays reflected wave directions on each facet of reflector.

### 3.5 The *ANGLES* key word

#### Description

This is mandatory and defines the angular range at which the far-field gain is required. This defines a uniformly sampled block of solid angle defined by a range of  $\theta$  and  $\phi$ .

The list below summarises its use.

Number of parameters	6
Key word requirement	Mandatory.
Number of occurrences	1.
File location	Anywhere within input file.
Parameter #1	Real number. Defines the first value of $\theta$ in degrees.
Parameter #2	Real number. Defines the $\theta$ increment in degrees.
Parameter #3	Integer. Defines the total number of $\theta$ values.
Parameter #4	Real number. Defines the first value of $\phi$ in degrees.
Parameter #5	Real number. Defines the $\phi$ increment in degrees.
Parameter #6	Integer. Defines the total number of $\phi$ values.

Note that we may set Parameter #3 and Parameter #6 to zero, in which case *ANGLES* does not specify any requested directions. Angles may also be requested using *ANGLECUT* any number of times (see below). If no angles are specified either by *ANGLES* or by *ANGLECUT* then the program will terminate with an error message.

### 3.6 The *ANGLECUT* key word

#### Description

This is optional. It may be used any number of times. It defines a single cut (angular sweep) about any great plane of the direction sphere. Data output from each *ANGLECUT* is appended sequentially to the output file in order of their appearance in the input file, after the data generated by the *ANGLES* definition. The output file (as of version 1.004 of the software, in which this tag is introduced) contains a field position 10 which outputs the scan angle  $\nu$ . See the previous documentation concerning the definitions of angles here.

The list below summarises its use.

Number of parameters	5
Key word requirement	Optional.
Number of occurrences	Any number.
File location	Anywhere within input file.
Parameter #1	Real number, $\theta_0$ , defining the value for $\theta$ of the centre point of the cut in degrees.
Parameter #2	Real number, $\phi_0$ , defining the value for $\phi$ of the centre point of the cut in degrees.
Parameter #3	Real number, $\eta_0$ , defining the angle of the cut in degrees, with respect to the direction $\hat{\theta}_0$ .
Parameter #4	Real number, $\Delta\nu$ , defining the angular increment along the cut, in degrees.
Parameter #5	Integer. Defines the total number of increments such that the sweep is made, $-n\Delta\nu \leq \nu \leq n\Delta\nu$ .

### 3.7 The *FREQS* key word

#### Description

This is mandatory and defines the frequency range for which predictions are required. Frequencies are entered in MHz.

The list below summarises its use.

Number of parameters	3
Key word requirement	Mandatory.
Number of occurrences	1.
File location	Anywhere within input file.
Parameter #1	Real number. Defines the first value of frequency in MHz.
Parameter #2	Real number. Defines the frequency increment in MHz.
Parameter #3	Integer. Defines the total number of frequency values.

### 3.8 The *FEEDCEN* key word

#### Description

This is mandatory and defines the phase centre origin of the excitation. For a plane wave this defines the reference point for which the phase is zero and defines a front/back reference to the reflector surface normal. For other feeds it defines the phase centre of the feed.

The list below summarises its use.

Number of parameters	3
Key word requirement	Mandatory.
Number of occurrences	1.
File location	Anywhere within input file.
Parameter #1	Real number. Defines the x-coordinate of phase centre in metres in global coordinate system.
Parameter #2	Real number. Defines the y-coordinate of phase centre in metres in global coordinate system.
Parameter #3	Real number. Defines the z-coordinate of phase centre in metres in global coordinate system.

### 3.9 The *FEEDROT* key word

#### Description

This is mandatory if a real feed is used, such as a dipole or horn. It must not be present if an incident plane wave is employed in order to avoid possible confusion, since a plane wave is defined with respect to the global coordinate system. For a feed, this defines the Euler rotation angles of the feed,  $\theta_f$ ,  $\phi_f$  and  $\psi_f$ . See earlier notes for a definition of these.

The list below summarises its use.

Number of parameters	3
Key word requirement	Special. Mandatory if a real feed is employed, otherwise must not be present.
Number of occurrences	1 or 0.
File location	Anywhere within input file.
Parameter #1	Real number. Defines the Euler rotation angle $\theta_f$ in degrees.
Parameter #2	Real number. Defines the Euler rotation angle $\phi_f$ in degrees.
Parameter #3	Real number. Defines the Euler rotation angle $\psi_f$ in degrees.

### 3.10 The *PLANEWAVE* key word

#### Description

One of several methods to illuminate the reflector and is logically considered as a feed type. This defines an incident plane wave of arbitrary polarisation and ellipticity. One and only one feed type must be used. A plane wave is specified by an incoming wave along the direction specified by  $\theta_p$  and  $\phi_p$ , with a polarisation defined by the angles  $\chi_\alpha$  and  $\chi_\beta$ . The former defines the angle of the principal axis of the polarisation ellipse with respect to the electric field vectors. When  $\chi_\beta = 0$ , this refers to the polarisation angle of a linearly polarised wave. When  $\chi_\alpha = 45$  degrees, the two states of circular polarisation are defined when  $\chi_\beta = \pm 90$  degrees. More generally, the plane wave is elliptically polarised with the electric vector amplitude taking the form,

$$\underline{E}_{inc} = \cos \chi_\alpha \hat{\underline{\theta}}_p + e^{j\chi_\beta} \sin \chi_\alpha \hat{\underline{\phi}}_p \quad (3-3)$$

When the *PLOTFILE* command is used, the direction of the plane wave is represented by an arrow through the phase centre point specified by *FEEDCEN*. The principal direction (shown as  $\cos \chi_\alpha \hat{\underline{\theta}}_p + \sin \chi_\alpha \hat{\underline{\phi}}_p$ ) is also illustrated through the phase centre point by an un-arrowed line.

The list below summarises its use.

Number of parameters	5
Key word requirement	Special. Represents one of several types of ‘feed’. One and only one ‘feed’ type must be present.
Number of occurrences	1 or 0.
File location	Anywhere within input file.
Parameter #1	Real number. Defines the incidence angle $\theta_p$ in degrees.
Parameter #2	Real number. Defines the incidence angle $\phi_p$ in degrees.
Parameter #3	Real number. Defines the polarisation angle $\chi_\alpha$ in degrees.
Parameter #4	Real number. Defines the ellipticity angle $\chi_\beta$ in degrees.

### 3.11 The *DIPOLE* key word

#### Description

One of several methods to illuminate the reflector and is logically considered as a feed type. This defines an elementary perfect (Hertzian) electric dipole radiator, situated at the phase centre point defined by *FEEDCEN*. The electric dipole is assumed to lie in the  $\hat{z}'$  direction. As of version 0.0.12 of the software, this takes no parameters though it is planned to generalise this dipole at some later time.

When the *PLOTFILE* command is used, the  $\hat{z}'$  direction of the dipole is represented by an arrow through the phase centre point specified by *FEEDCEN*.

The list below summarises its use.

Number of parameters	0
Key word requirement	Special. Represents one of several types of 'feed'. One and only one 'feed' type must be present.
Number of occurrences	1 or 0.
File location	Anywhere within input file.

### 3.12 The *RECTHORN* key word

#### Description

One of several methods to illuminate the reflector and is logically considered as a feed type. This defines a rectangular aperture horn with a cosine electric field distribution on the aperture. The feed horn is defined by the widths of the aperture in the  $\hat{x}'$  and  $\hat{y}'$  directions, as well as offset phase distances that define the distance of the phase centre to the aperture associated with  $\hat{x}'$  and  $\hat{y}'$  directions. Also required is a meshing interval in fractions of a wavelength and a 'type' designator to define the manner of polarisation of the horn.

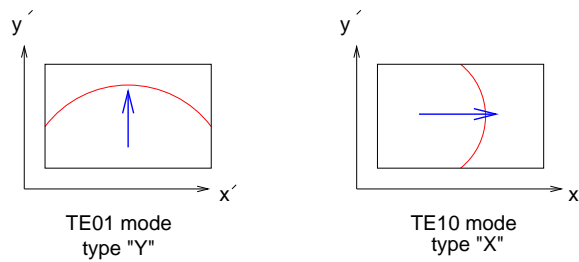
Since the phase centre of the source may not exist at a single point in space (i.e. if the offset phase distances are different) and since the local phase centre of the aperture field may not coincide with the apparent phase centre as seen by the reflector in the intermediate or far field of the horn, it is generally best to assume the physical location of the phase centre defined by *FEEDCEN* lies close to the centre of the aperture of the horn. The offset phase distances required here are principally required to accurately define the aperture field distribution from which the field at the reflector is calculated.

The meshing interval should typically lie between a value of 0.2 and 0.01, depending on the accuracy required and the computation time penalty. We have usually found that a sampling value of 0.1 is quite adequate.

The 'type' designator is a character 'X', 'Y', 'LCP' or 'RCP'. 'X' implies the electric field on the aperture is aligned in the  $\hat{x}'$  direction. 'Y' implies the electric field on the aperture is aligned in the  $\hat{y}'$  direction. 'LCP' and 'RCP' imply that the feed is left or right circularly

polarised. Note that the main reflector reverses the sense of circularity of the polarisation of the radiated field.

It is assumed here that only a TE01 and a TE10 mode can be excited. When either one or the other is present (but not both), the 'type' is designated by 'Y' or 'X' respectively. The LCP and RCP types involve a suitably phased linear combination of both. Figure 3-3 shows the excitation type with respect to the local feed coordinate system.



*Figure 3-3: Aperture excitation types for a rectangular horn*

When the *PLOTFILE* command is used, the  $\hat{z}'$  direction of the feed and its aperture are illustrated.

The list below summarises its use.

Number of parameters	6 or 8
Key word requirement	Special. Represents one of several types of ‘feed’. One and only one ‘feed’ type must be present.
Number of occurrences	1 or 0.
File location	Anywhere within input file.
Parameter #1	Real number. Defines the aperture width in $x'$ direction in metres.
Parameter #2	Real number. Defines the aperture width in $y'$ direction in metres.
Parameter #3	Real number. Defines the distance to the local phase centre for the TE01 (type ‘Y’) mode.
Parameter #4	Real number. Defines the distance to the local phase centre for the TE10 (type ‘X’) mode.
Parameter #5	Character string. Defines the ‘type’ designator, as described above.
Parameter #6	Real number. Defines the aperture sampling interval as a wavelength fraction.
Parameter #7	Character string. Optional. If present, defines the name of an output file defining the aperture electric field on the sampling grid.
Parameter #8	Character string. Optional. Must be present if parameter #7 is present. If present, defines the name of an output file defining the aperture magnetic field on the sampling grid.

Note that parameters #7 and #8 (included as of version 0.0.14) are optional. If present they should contain no separators and define the name of output files containing the transverse electric and magnetic fields on the aperture. The components of the field perpendicular to the aperture are defined as zero. This file is in the same format as a CST file [4] and may, for example, be used in tests of the *FEEDFILE* command.

### 3.13 The *GEOMFILE* key word

#### Description

This key word defines the name of a secondary file required by the program and is mandatory. There are two uses of this command. The first parameter defines the name of a “points and joins” file. The second defines a permission and the mode of operation. This takes the form of a two character string; either ‘RO’ or ‘RW’. In ‘RO’ mode, the defined file can only be read. In ‘RW’ mode, the designated file can be read and written to.

The effect of this command depends on whether the *SURFACE* and *BOUNDARY* commands are present. *SURFACE* and *BOUNDARY* commands must either both be present or neither must be present. If both are present the points-and-joins file will be written to if ‘RW’ is designated or there is no effect if ‘RO’ is designated (i.e. this command is ignored). In both cases, the reflector geometry is defined by the *SURFACE* and *BOUNDARY* commands and not the points-and-joins file.

If neither *SURFACE* or *BOUNDARY* commands are present, the software takes the reflector geometry defined by the points-and-joins file. In this case the permission must be set to ‘RO’.

The required format of the points and joins file is shown in the following example (note that the node coordinates are specified in metres):

Number of nodes:

16

Node coordinates:

1	0.80680E-01	0.99296E-04	0.48644E-01
2	0.34082E-01	0.69546E-04	0.71018E-01
3	-0.20870E-01	0.24396E-04	0.73532E-01
4	-0.79174E-01	0.13780E-02	0.49868E-01
5	-0.10600E+00	0.30000E-02	0.41399E-05
6	-0.80663E-01	0.15101E-02	-0.48659E-01
7	-0.31787E-01	0.19570E-03	-0.71548E-01
8	0.25243E-01	-0.46067E-05	-0.72842E-01
9	0.80294E-01	0.10700E-03	-0.48964E-01
10	0.10600E+00	0.22617E-03	0.43846E-13
11	0.15782E-01	0.48883E-04	0.30572E-01
12	-0.35072E-01	0.80834E-04	0.21060E-01
13	-0.34648E-01	0.88332E-04	-0.21409E-01
14	0.15777E-01	0.48921E-04	-0.30573E-01
15	0.46000E-01	-0.45332E-05	0.19028E-13
16	0.00000E+00	0.41190E-04	0.00000E+00

Number of facet elements:

20

Element reference list:

1	1	2	11
2	2	3	11
3	3	4	12
4	4	5	12
5	5	6	13
6	6	7	13
7	7	8	14
8	8	9	14
9	9	10	15
10	10	1	15
11	11	12	3
12	12	13	5
13	13	14	7
14	14	15	9
15	15	11	1
16	11	12	16
17	12	13	16
18	13	14	16
19	14	15	16
20	15	11	16

The first list defines the node coordinates of the reflector in metres and assigns a node number to each point. The second list defines the triangles by reference to the node such that each triangle is defined by three nodes.

The list below summarises the command use.

Number of parameters	2
Key word requirement	Mandatory.
Number of occurrences	1
File location	Anywhere within input file.
Parameter #1	Character string containing no separators. Defines the name of the points and joins file.
Parameter #2	Character string. Defines the permission. This must be either 'RO' or 'RW'.

### 3.14 The *SURFACE* key word

#### Description

This is mandatory if the reflector geometry is not defined by the *GEOMFILE* command. If present, it must also be accompanied by the *BOUNDARY* command. The first parameter defines the type of surface. Subsequent parameters depend on the type of surface.

Currently (version 0.1.02) there are five types of permitted surface; a plane surface (*PLANE*), a parabolic surface (*PARABOLOID*), a distorted parabolic surface (*PARABDISTORT*), a shaped parabolic surface (*VARIPARAB1*) and an extruded one-dimension cubic spline surface (*SPLINE1D*). The boundary of all these surfaces is defined by the *BOUNDARY* command.

#### The *PLANE* surface

This surface is a simple flat mirror whose normal is defined by the vector  $\underline{n}_r = (n_x, n_y, n_z)$  and a point  $\underline{p} = (p_x, p_y, p_z)$  in the plane.

#### The *PARABOLOID* surface

This surface is a paraboloid of revolution. It is defined by its focal length,  $f_l$ , and the coordinates of the focal point,  $\underline{r}_f = (x_f, y_f, z_f)$ . It is assumed that the paraboloid is always orientated such that its axis of revolution is parallel to the y-axis.

#### The *PARABDISTORT* surface

This surface is a paraboloid of revolution, as above, with an additive deviation function designated by a type (specified by an integer) and a filename containing the deviation parameters associated with the type.

#### The *VARIPARAB1* surface

This surface represents a distorted paraboloid of revolution intended to permit a profiled beam in one angular direction (e.g. an approximation to a  $\text{cosec}^2\theta$  distribution in the  $\theta$  direction) while maintaining a narrow beam in the orthogonal  $\phi$  direction. There are any number of ways in which this can be done. Here we will assume that the reflector surface function  $y_s(x, z)$  is given by,

$$y_s(x, z) = \alpha(z)y_p(x, z) \quad (3-4)$$

where  $y_p(x, z)$  is a paraboloid of revolution about the y-axis, as defined by the *PARABOLOID* option. The function  $\alpha(z)$  defines the distortion. We aim for a function such that  $\alpha(z)$  is close to one for  $z > 0$ , with progressive deviation for smaller  $z$ . We choose,

$$\alpha(z) = 1 - (\alpha_0 + \alpha_1 z + (\alpha_2 z)^2 + \dots)e^{-\beta z} \quad (3-5)$$

where  $\alpha_0$  is dimensionless and  $\beta, \alpha_1, \dots, \alpha_n$  have units of  $m^{-1}$ . We will assume a total of  $N$  such  $\alpha_i$  parameters.

### The SPLINE1D surface

This surface assumes a general reflector surface given by,

$$y_s(x, z) = C(x) \quad (3-6)$$

which is independent of  $z$  and where  $C(x)$  is defined by a one-dimensional cubic spline function whose control points are provided by an auxiliary input file. We assume a ‘natural’ cubic spline with control points that must span the range of values required by the *BOUNDARY* defined boundary. The required input file name is a parameter, and takes the format (coordinates specified in metres) :

```
itot
x(1)    y(1)
x(2)    y(2)
.        .
.        .
x(itot) y(itot)
```

where  $x(1) < x(2) < \dots < x(itot)$ . The algorithm for defining  $C(x)$  is given in pp119 of [3]. Such a surface may be regarded as an extrusion of a cubic curve along the  $z$ -axis, truncated by the *BOUNDARY*. The list below summarises the command use.

Number of parameters	At least 2.
Key word requirement	Special. Mandatory if <i>BOUNDARY</i> command present.
Number of occurrences	1 or 0.
File location	Anywhere within input file.
Parameter #1	Character string. This defines the surface type and must be either <i>PLANE</i> , <i>PARABOLOID</i> , <i>PARABDISTORT</i> , <i>VARIPARAB1</i> or <i>SPLINE1D</i> .

Further parameters depend on the surface type.

**If Parameter #1 = PLANE** then,

Parameter #2	Real number. Defines the x-component of the surface normal in metres.
Parameter #3	Real number. Defines the y-component of the surface normal in metres.
Parameter #4	Real number. Defines the z-component of the surface normal in metres.
Parameter #5	Real number. Defines the x-component, $p_x$ , of a surface point in metres.
Parameter #6	Real number. Defines the y-component, $p_y$ , of a surface point in metres.
Parameter #7	Real number. Defines the z-component, $p_z$ , of a surface point in metres.

**If Parameter #1 = PARABOLOID** then,

- Parameter #2 Real number. Defines  $f_l$ , the focal length of parabola in metres.
- Parameter #3 Real number. Defines the x-component of the focal point in metres.
- Parameter #4 Real number. Defines the y-component of the focal point in metres.
- Parameter #5 Real number. Defines the z-component of the focal point in metres.

**If Parameter #1 = PARABDISTORT** then,

- Parameter #2 Real number. Defines  $f_l$ , the focal length of parabola in metres.
- Parameter #3 Real number. Defines the x-component of the focal point in metres.
- Parameter #4 Real number. Defines the y-component of the focal point in metres.
- Parameter #5 Real number. Defines the z-component of the focal point in metres.
- Parameter #6 Integer number. Specifies the distortion type,  $ndistort$ .
- Parameter #7 Character string. Specifies the name of the file containing the distortion parameters,  $distortfilename$ .

Currently (version 1.02) only options are  $ndistort = 0$ , where there is no distortion applied, functionally equivalent to the use of Parameter #1 = PARABOLOID, or  $ndistort = 1$ . When  $ndistort = 1$ , the coordinates of the reflector surface are defined by,

$$y_s(x, z) = y_p(x, z) + d(r, \theta) \quad (3-7)$$

where,

$$x = r \cos \theta \quad , \quad z = r \sin \theta \quad (3-8)$$

and  $d(r, \theta)$  is a deviation function defined by,

$$d(r, \theta) = R(r)\Theta(\theta) \quad (3-9)$$

with

$$R(r) = \rho_1 r^2 + \rho_2 r^4 + \dots + \rho_N r^{2N} \quad (3-10)$$

$$\Theta(\theta) = a_0 + \sum_{i=1}^M (a_i \cos(i\theta) + b_i \sin(i\theta)) \quad (3-11)$$

for real coefficients  $\rho_i$ ,  $a_i$  and  $b_i$ .

The named file *distortfilename* must be present or an error is flagged. When present, it must take the form (comments optional):

```

M      comment. Integer  $M \geq 0$ 
 $a_0$     comment.
 $a_1$ 
.... ..
.... ..
 $a_M$     comment. Last of the  $a_i$  coefficients.
 $b_1$     comment.
.... ..
.... ..
 $b_M$     comment. Last of the  $b_i$  coefficients.
#      comment. This must be a hash designator to separate file parts.
N      comment. Integer  $N \geq 1$ 
 $\rho_1$    comment.
.... ..
.... ..
 $\rho_N$    comment. Last of the  $\rho_i$  coefficients.
```

For example, for a circular dish of radius  $r_0$  we may employ  $N = 2$ , specify  $\rho_1 = 1/r_0^2$  and  $\rho_2 = -1/r_0^4$  so that there is no deviation at the rim and maximum deviation when  $r = r_0/\sqrt{2}$ . With this special form,  $R(r)$  is dimensionless and the coefficients  $a_i$  and  $b_i$  take units of distance. All units of distance are specified in metres.

**If Parameter #1 = VARIPARAB1** then,

- Parameter #2 Real number. Defines  $f_l$ , the focal length of unmodified parabola  $y_p(x, z)$  in metres.
- Parameter #3 Real number. Defines the x-component of the focal point of  $y_p(x, z)$  in metres.
- Parameter #4 Real number. Defines the y-component of the focal point of  $y_p(x, z)$  in metres.
- Parameter #5 Real number. Defines the z-component of the focal point of  $y_p(x, z)$  in metres.
- Parameter #6 Integer  $\geq 0$ . The number,  $N$ , of additional  $\alpha_i$  parameters (for  $1 \leq i \leq N$ ).
- Parameter #7 Real number. The exponential decay parameter,  $\beta$ , in  $m^{-1}$ .
- Parameter #8 Real number. The first mandatory dimensionless parameter,  $\alpha_0$ .
- Parameter #9 Real numbers. These are the values  $\alpha_i$  in units of  $m^{-1}$ .
- to #(N+8) If  $N = 0$  these parameters are not specified.

**If Parameter #1 = SPLINE1D** then,

- Parameter #2 Character string containing no separators. This defines the name of the input file specifying the cubic spline control points as defined previously.

### 3.15 The *BOUNDARY* key word

#### Description

This is mandatory if the reflector geometry is not defined by the *GEOMFILE* command. If present, it must also be accompanied by the *SURFACE* command. The first parameter defines the type of boundary. Subsequent parameters depend on the type of boundary.

Currently (version 0.0.12) there are two types of permitted surface; a rectangular boundary (*RECTANGLE*) and an elliptical boundary *ELLIPSE*. The boundary command defines a uniformly meshed flat bounded rectangle or ellipse, defined perpendicular to the  $y$ -axis. The triangles and nodes of this flat mesh are then projected along the  $y$ -axis on to the surface defined by the *SURFACE* command.

The list below summarises the command use.

Number of parameters	At least 2.
Key word requirement	Special. Mandatory if <i>SURFACE</i> command present.
Number of occurrences	1 or 0.
File location	Anywhere within input file.
Parameter #1	Character string. This defines the boundary type and must be either <i>RECTANGLE</i> or <i>ELLIPSE</i> .

Further parameters depend on the surface type.

**If Parameter #1 = *RECTANGLE* then,**

Parameter #2	Real number. The rectangle width, $w_x$ , in metres defined parallel to the $x$ -axis prior to rotation.
Parameter #3	Real number. The rectangle height, $w_z$ , in metres defined parallel to the $z$ -axis prior to rotation.
Parameter #4	Real number. $x$ -coordinate of rectangle centre, $x_c$ .
Parameter #5	Real number. $z$ -coordinate of rectangle centre, $z_c$ .
Parameter #6	Real number. Rotation angle, $\theta_r$ , of rectangle about its centre, rotated about a vector parallel to the $y$ -axis. Defined in degrees.
Parameter #7	Integer. Number of meshing divisions, $n_x$ , along the $x$ -axis.
Parameter #8	Integer. Number of meshing divisions, $n_z$ , along the $z$ -axis.

See figure 3-4 below.

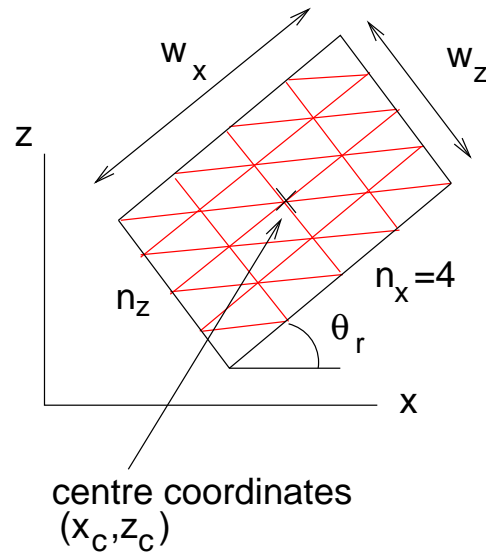


Figure 3-4: Boundary rectangle and its projection

If **Parameter #1 = ELLIPSE** then,

- Parameter #2 Real number. Ellipse semi-major/minor axis along x-direction, before rotation. In metres.
- Parameter #3 Real number. Ellipse semi-major/minor axis along z-direction, before rotation. In metres.
- Parameter #4 Real number.  $x$ -coordinate of ellipse centre,  $x_c$ .
- Parameter #5 Real number.  $z$ -coordinate of ellipse centre,  $z_c$ .
- Parameter #6 Real number. Rotation angle,  $\theta_r$ , of ellipse about its centre, rotated about a vector parallel to the  $y$ -axis. Defined in degrees.
- Parameter #7 Real number. Target mesh size in metres.

Note that the target mesh size is used as a meshing criterion and is not precisely met. Meshing method is hard-wired into the software. This can be changed (and other hard-wired options used), but the default method divides the ellipse into concentric shells together with a central core. Each shell is uniformly divided into triangles to the target mesh size. The final remaining core assumes a single node at the centre, so that triangles here may be rather smaller.<sup>1</sup>

<sup>1</sup>Details of the method are not reported here.

### 3.16 The *BLOCKAGE* key word

#### Description

This command is optional. If present it must occur only once and describes the nature of a blockage to the main reflector. Currently (software version 0.0.12) it takes two forms. The first form is where the feed is responsible for the blockage (*FEED* designation), assuming that it is only the feed aperture that is responsible for the blockage. The second form (*NEW* designation) requires the user to specify a file name containing a points-and-joins file with the same format as specified in defining the reflector. This allows an arbitrary blockage to be defined.

The list below summarises the command use.

Number of parameters	At least 1.
Key word requirement	Optional.
Number of occurrences	1 or 0.
File location	Anywhere within input file.
Parameter #1	Character string. This defines the feed type and must be either <i>FEED</i> or <i>NEW</i> .

**If Parameter #1 = FEED** then there are no further parameters.

**If Parameter #1 = NEW** then

Parameter #2	Character string containing no separators. This is the file name of the points-and-joins file specifying the blockage.
--------------	--

### 3.17 The *NEARFIELD* key word

#### Description

This command is optional. If present it defines a rectangular region in space where the near fields are to be evaluated. Various options are also provided to provide an alternative method to calculate the effect of the blockage to the far field and/or to determine the far field from the calculated near field.

Firstly, it is assumed that the near field is to be specified on the plane  $y = y_0$  on a rectangle defined by  $x_{min} \leq x \leq x_{max}$  and  $z_{min} \leq z \leq z_{max}$ .

Secondly, if the *shadow mask* option is flagged as 'True', the geometric rays from the reflector are traced through the blockage on to the  $y = y_0$  plane. These are the rays previously discussed when defining the effect of the blockage. Now, however, *all* the currents are evaluated on the reflector including the ones which (in the basic method) are taken as non-contributory if the respective facets were taken as being in shadow. In this option, it is the plane  $y = y_0$  plane which defines the existence of shadow. Thus, if any shadowing rays intersect the plane  $y = y_0$  for which  $x_{min} \leq x \leq x_{max}$  and  $z_{min} \leq z \leq z_{max}$  then the fields there are assumed zero. Given a, possibly masked, near field region the far field may then be computed. This is performed if the *far-field* flag is set 'True'. Figure 3-5 shows the near field region and the effect of shadowing on the far field using this alternative method.

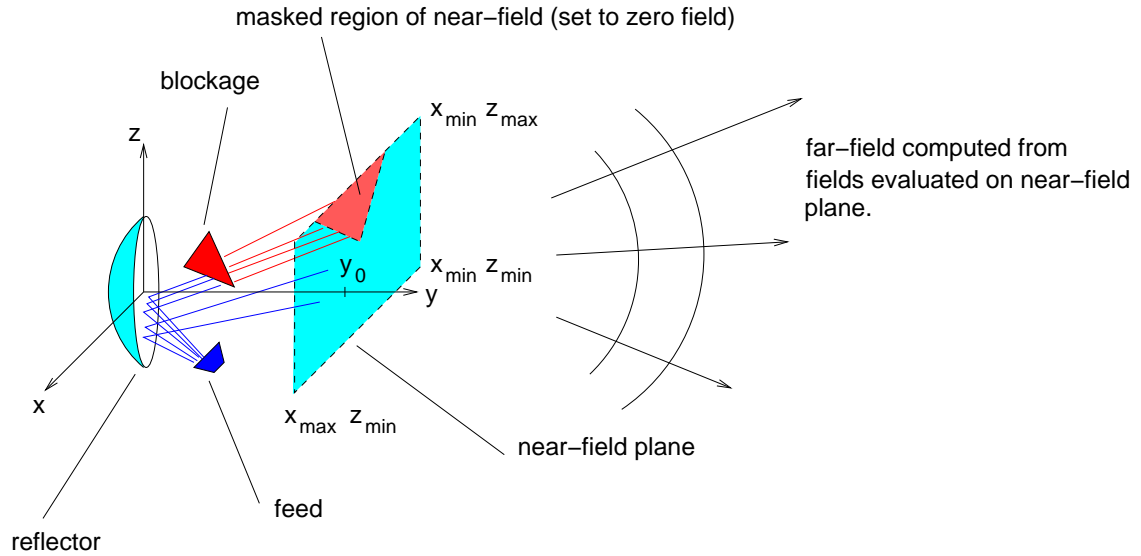


Figure 3-5: Near field region, shadow masking and far field

The method of near-to-far field transformation is similar to the one employed for evaluating the far-field from the aperture of a feed. The method uses (2-2) where,

$$\begin{aligned} \underline{J}_e &= \hat{n} \times \underline{H}(\underline{r}') \\ \underline{J}_m &= -\hat{n} \times \underline{E}(\underline{r}') \end{aligned} \quad (3-12)$$

where  $\underline{H}(\underline{r}')$  and  $\underline{E}(\underline{r}')$  are the near fields evaluated on the plane  $y = y_0$ . Note that the sampling must be sufficiently fine over the rectangular region (typically with a sampling interval  $\lesssim 0.2$  wavelengths) and the rectangle must be large enough such that the fields outside the rectangle are small enough to ignore. When no blockages are present, this method of computing the far-field is inferior to the main method.

The list below summarises the command use.

Number of parameters	11.
Key word requirement	Optional.
Number of occurrences	1 or 0.
File location	Anywhere within input file.
Parameter #1	Real number. This defines the position of evaluation plane, $y_0$ , in metres.
Parameter #2	Real number. Minimum x-coordinate of near-field region, $x_{min}$ in metres.
Parameter #3	Real number. Maximum x-coordinate of near-field region, $x_{max}$ in metres.
Parameter #4	Integer. Number of evaluation points, $n_x$ , between $x_{min}$ and $x_{max}$ .
Parameter #5	Real number. Minimum z-coordinate of near-field region, $z_{min}$ in metres.
Parameter #6	Real number. Maximum z-coordinate of near-field region, $z_{max}$ in metres.
Parameter #7	Integer. Number of evaluation points, $n_z$ , between $z_{min}$ and $z_{max}$ .
Parameter #8	Character. This represents the shadow mask flag. If this is 'T', then a blockage is assumed, defined by the <i>BLOCKAGE</i> option. If this is 'F', no account is take of blockage effects.
Parameter #9	Character. This represents the far-field flag. If this is 'T', then the far-field is calculated from the near-field region, with or without masked fields. If this is 'F' then no far-fields are calculated using this method.
Parameter #10	Character string with no separators. This is the name of the near-field file to be created.
Parameter #11	Character string with no separators. This is the name of the far-field file to be created.

### 3.18 The *GRASPOUT* key word

#### Description

This command is optional and constructs a geometry .sfc file suitable for input into GRASP [1]. The file only contains the reflector nodes with coordinates offset by the specified  $x$ ,  $y$  and  $z$  offsets,  $x_0$ ,  $y_0$ ,  $z_0$  defined here. GRASP uses this “scatter profile” to generate its own curved surface through these points. This file may only be used if a full version of GRASP is available for analysis.

Number of parameters	4
Key word requirement	Optional.
Number of occurrences	1 or 0.
File location	Anywhere within input file.
Parameter #1	Character string containing no separators. This defines the name of the GRASP .sfc file.
Parameter #2	Offset coordinate $x_0$ in metres.
Parameter #3	Offset coordinate $y_0$ in metres.
Parameter #4	Offset coordinate $z_0$ in metres.

### 3.19 The *FILEREFL* key word

#### Description

This command is optional and is intended mostly for diagnostics. It provides the incident electric and magnetic field components evaluated at the nodal points of the reflector surface. There are three coordinate options. In the first (designation *CART\_REFL*), the fields are defined in the global cartesian coordinate system. In the second (designation *SPH\_FEED*), the fields are defined in spherical coordinates in the coordinate system of the feed. In the third (designation *CART\_FEED*), the fields are defined in cartesian coordinates in the coordinate system of the feed.

Number of parameters	4
Key word requirement	Optional.
Number of occurrences	1 or 0.
File location	Anywhere within input file.
Parameter #1	Character string containing no separators. Name of the electric field output file.
Parameter #2	Character string containing no separators. Name of the magnetic field output file.
Parameter #3	Character string. Coordinate options <i>CART_REFL</i> , <i>SPH_FEED</i> or <i>CART_FEED</i> .

### 3.20 The *FEEDFILE* key word

#### Description

One of several methods to illuminate the reflector and is logically considered as a feed type. This defines a user-specified feed aperture distribution, whose format is that of a CST [4] output file. It is included as of version 0.0.13 of the software.

An aperture distribution is specified by a regular rectangular grid of points where the fields are specified. There are two options: either the electric field only is specified or both the electric and magnetic fields. In both cases, the  $z'$ -components of the fields perpendicular to the aperture are not required although they are specified in the input files for consistency with CST [4] format.

Let us assume the aperture lies in the  $x', y'$  plane of the feed coordinate system. When only the (transverse) electric field is specified The transverse magnetic field is approximated from the tangential electric fields  $(E_x, E_y)$  via,

$$\underline{H} \approx H_x \underline{\hat{x}}' + H_y \underline{\hat{y}}' \quad (3-13)$$

where

$$H_x \approx -f_0 Z_0 E_y \quad H_y \approx f_0 Z_0 E_x \quad (3-14)$$

where  $f_0$  is a power scaling factor taken to ensure that the power radiated by the aperture is unity. This approximation is valid if the fields are locally planar and is usually considered good in the apertures of horns, provided the flair angle is not too large.

If both the (transverse) electric and magnetic fields are available this approximation is not required, though we still calculate the component of the (real part of the) Poynting vector normal to the aperture to normalise total power radiated by the aperture to unity.

The component of the reactive part of the Poynting vector is also computed as a diagnostic. If only the (transverse) electric field is available this component is estimated as exactly zero. If both electric and magnetic fields are available it will in general be non-zero. If so, the reactive component should be small compared to the radiated part or else the program will issue a warning and the supplied aperture fields should be considered as poor.

The specified aperture files take the same format. This consists of a list of rows where the first three entries in a row specify the  $x', y'$  and  $z'$  coordinates in millimetres (the  $z'$  coordinates are not used except for diagnostics; i.e. the field is assumed to be given on the plane  $z' = 0$ ).<sup>2</sup> The next three specify the  $x', y'$  and  $z'$  components of the real part of the electric field (electric field file) or magnetic field (magnetic field file). The next three specify the  $x', y'$  and  $z'$  components of the imaginary parts of the fields. This is consistent with CST output files.

---

<sup>2</sup>Note that the units of mm are for compatibility with CST export files where, for microwave applications, units are specified in mm. This is in contrast with the units of distance elsewhere in the software which are specified in metres

When using an exported CST output file, employ a text (.txt) export format and ensure distance units are in millimetres (this is the CST default condition). Also ensure that the plane on which the fields are specified is transverse to the CST z-axis, for consistency with the above feed coordinate system. Finally, remove (delete) the header lines in the text file containing the ascii description of the numbers. There are (CST versions 6 to 10) two such header lines.

When only the electric field file is supplied, scaling of the electric field components is unimportant since the program will re-normalise to assume 1 Watt is radiated. When both field files are supplied there can be an arbitrary common scaling factor, but this must be common to both files. A regular grid is assumed. If the program detects the specified coordinates do not form a regular grid, or there is a difference in the specified coordinates for the electric and magnetic field data, an error is flagged.

In theory, the use of both electric and magnetic fields should result in greater accuracy which is important for describing cross-polar performance and side-lobes. However, as of version 1.003 (May 2010), we have not performed exacting validation to test this assertion.

The list below summarises the use of this command.

Number of parameters	5 or 6
Key word requirement	Special. Represents one of several types of 'feed'. One and only one 'feed' type must be present.
Number of occurrences	1 or 0.
File location	Anywhere within input file.
Parameter #1	Integer. This must be 1 or 2. If 1, then only the electric field is supplied. If 2, then both electric and magnetic field files are required.
Parameter #2	Name of the required electric field data file (CST format)

**If parameter #1=1, then**

Parameter #3	Frequency in MHz
Parameter #4	Display scaling distance in metres ( $x'$ direction)
Parameter #5	Display scaling distance in metres ( $y'$ direction)

**If parameter #1=2, then**

Parameter #3	Name of the required magnetic field data file (CST format)
Parameter #4	Frequency in MHz
Parameter #5	Display scaling distance in metres ( $x'$ direction)
Parameter #6	Display scaling distance in metres ( $y'$ direction)

Parameter #3(4) must be the same as the frequency (only a single frequency is permitted here) specified by the *FREQS* command. This is required explicitly as a check since the

CST file does not specify the frequency at which the data is generated. If the frequency is not correct the program terminates. Parameters #4(5) and #5(6) are only used for illustration purposes for use by *PLOTFILE*. This is because the aperture file is read in after the information is defined for generating the POSTSCRIPT files and so the aperture coordinates available here cannot easily be used for this purpose. It is logically preferred to use different illustration parameters until or unless the software is significantly changed.

### 3.21 The *FARPOL* key word

Introduced in version 1.003 of the software, this permits a rotation of the polarisation base of the far-field and defines a value for the polarisation rotation angle  $\zeta_s$ . The key word is optional but if present must occur only once. If not present is assumed that  $\zeta_s = 0$ .

The list below summarises the use of this command.

Number of parameters	1
Number of occurrences	0 or 1
File location	Anywhere within input file.
Parameter # 1	Real number. This is the angle $\zeta_s$ in degrees.

## 4 References

- 1 *TICRA report ed. K. Pontoppidan* ‘Technical description of GRASP 8’, version 8.2.5, TICRA, March 2002.
- 2 *J. Synge, B. Griffith* ‘Principles of Mechanics’, McGraw-Hill, 1970.
- 3 *W.H.Press, W.T.Vetterling, S.A.Teukolsky, B.P.Flannery* ‘Numerical Recipes in Fortran’, Cambridge University Press, 1986.
- 4 *CST EM simulation software*. See web site [www.cst.com](http://www.cst.com).
- 5 *S. Silver* ‘Microwave antenna theory and design’, McGraw-Hill, 1949.